COSC 4330 SECOND MIDTERM OCTOBER 26, 2006

This exam is **closed book**. You can have **one** page of notes. UH expels cheaters.

1. Which of the following statements are ***true*** or ***false*** (2 points) and ***why***? (3 points)

a) A ***non-blocking*** send returns as soon as the destination process has received the message that was sent.

FALSE, a ***non-blocking*** send returns as soon as message has been accepted for delviery by the kernel of the sender machine.

b) All semaphores should always be initialized to one.

FALSE, a semaphore can be initilaized to any value greater than or equal to zero. (Think of the solution to the producer/consumer problem discussed in class.)

c) Peterson's algorithm for mutual exclusion assumes the existence of a test-and-set instruction.

FALSE, Peterson's algorithm for mutual exclusion does not assume the existence of any special instruction.

d) The round-robin policy will ***never*** cause process starvation.

TRUE, all processes have the same priority.

e) The all or nothing semantics guarantees that all remote procedure calls will be executed at most once.

FALSE, the all or nothing semantics guarantees that all remote procedure calls will be completely executed (all) or not at all (nothing), excluding any possibility of partial execution or multiple executions.

f) UNIX sockets are an example o**f *public mailboxes.***

FALSE, UNIX sockets are an example of private mailboxes as they are attached to a single process.

2. Consider the instruction **`TSET R7, LOCK`** and assume it is used to ensure mutual exclusion within a critical section. Assuming that the variable **`LOCK`** can only be equal to zero or one. what are the two possible values for **`R7`** after the instruction is executed and their meanings? (2×5 points)

a) If R7 equals _0_ then the process can enter the critical section.

b) If R7 equals _1_ then the process cannot enter the critical section.

3. An electronics store has all shoppers waiting in a single waiting line for one of their twenty check-out clerks. After they complete their purchases, the shoppers must wait again to have their bags inspected

by one of the two inspectors standing at the door. Add the proper semaphore calls to represent this behavior. (20 points).

```
semaphore check_out = ____20____;
semaphore inspection = ___2_____;
shopper () {
        shop();
        _ P(&checkout); ____________________________________________
        pay();
        _ V(&checkout); P(&inspection); ____________________________
        show_bag();
        _ V(&inspection); ___________________________________________
} // shopper
```

4. What is the major advantage of

a) Datagrams over streams. (5 points)

_ Datagrams does not require any setup before transmitting a message. _____

b) Streams over datagrams. (5 points)

_ All messages will be received in order without lost messages or duplicates. _____

5. Consider the following solution to the mutual exclusion problem and explain when it fails (5 points) and what happens then. (5 points)

```
shared int occupied[2] = {0, 0}; // global variable

void enter_region(int pid) {  // pid will always be 0 or 1
     while (occupied[1 - pid]); // busy wait
     occupied[pid] = 1; // reserve
} // enter_region

void leave_region(int pid) {
     occupied[pid] = 0;
} // leave_region
```

When _ two processes try to enter the critical region in lockstep _____

_X_ It denies mutual exclusion . *(because we test before reservring!)*

___ It denies access to the critical section. *(this only happens if we reserve before testing and have no tie-breaking rule!)*

6. What is the major disadvantage of ***busy waits***? (5 points)

Busy waits create many context switches, thus wasting CPU cycles and slowing down the progress of the process inside the critical section.

7. Consider the System V Release 4 scheduler. (3×5 points). When should it:

a) ***Increase*** the priority of a process?

- When the process returns from the waiting state to the ready queue.
- When the process has been in the ready queue for more than ts_maxwait.

b) ***Decrease*** that priority?

- When the process has been preempted from the CPU for having exceeded its current time slice.

(*The number of points allocated to the question was an indication that we had to consider three cases!)*